Digital Logic Design

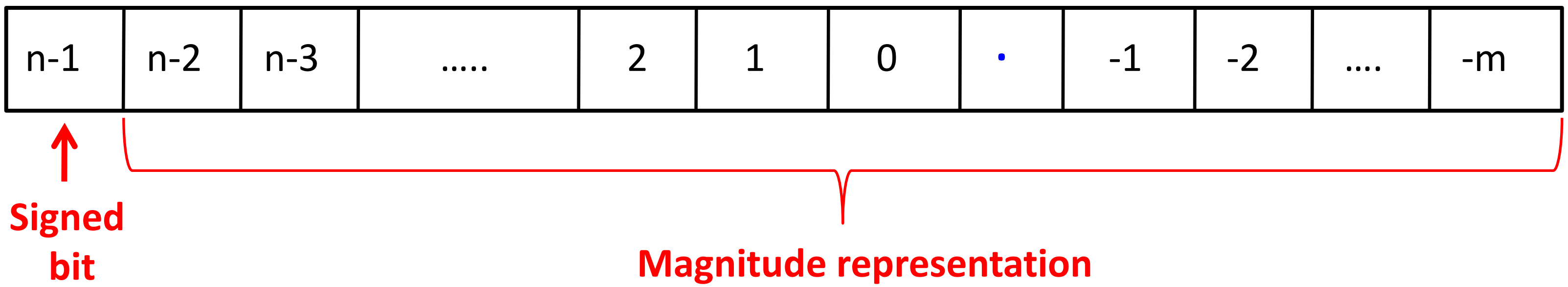
Hajar Falahati

Department of Computer Engineering
IRAN University of Science and Technology

hfalahati@iust.ac.ir

Signed Numbers Representation

Iran University of Science & Technology
IUST



- Let $N = (a_{n-1} \ldots a_0)_2$
  - If $N \geq 0$, it is represented by $(0a_{n-1} \ldots a_0)_2$
  - If $N < 0$, it is represented by $[0a_{n-1} \ldots a_0]_2$
    - $[N]_2 = 2^n - (N)_2$

Outline

- Signed and Unsigned Numbers
  - Sign magnitude
  - 2's complement
  - 1's complement
- Carry and overflow

Carry- Overflow

Radix Complement Arithmetic

- Suppose we have two 8-bit number
- 181 + 75
- $(10110101)_2 + (01001011)_2$

$$
\begin{array}{r}
181 \\
+\quad 75 \\
\hline
256
\end{array}
\qquad\qquad
\begin{array}{rcccccccc}
 & & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\
+ & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\
\hline
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}
$$

Overflow Condition

- Result of an operation falls outside the range
  - Fixed number of sum bits
  - Result is **not valid**
    - Adding two positive numbers and the sum is negative
    - Adding two negative numbers and the sum is positive

- Consider 3 cases
  - $A = B + C$,
  - $A = B - C$,
  - $A = -B - C$,

  (where $B \geq 0$ and $C \geq 0$.)

Overflow Condition: Case 1

- A = B + C
  - $(A)_2 = (B)_2 + (C)_2$
  - If $A > 2^{n-1} - 1$ (overflow)
    - It is **detected** by the n^{th} **bit**, which is set to **1**.

- Example: $(7)_{10} + (4)_{10}$ = ? using 5-bit two's complement arithmetic.
  - $+(7)_{10} = +(0111)_2 = (0, 0111)_{2cns}$
  - $+(4)_{10} = +(0100)_2 = (0, 0100)_{2cns}$
  - $(0, 0111)_{2cns} + (0, 0100)_{2cns} = (0, 1011)_{2cns} = +(1011)_2 = +(11)_{10}$
  - No overflow

- Example: $(9)_{10} + (8)_{10}$ = ?
  - $+(9)_{10} = +(1001)_2 = (0, 1001)_{2cns}$
  - $+(8)_{10} = +(1000)_2 = (0, 1000)_{2cns}$
  - $(0, 1001)_{2cns} + (0, 1000)_{2cns} = (1, 0001)_{2cns}$
  - Overflow

Overflow Condition: Case 2

- $A = B - C$
  - $(A)_2 = (B)_2 + (-(C)_2) = (B)_2 + [C]_2 = (B)_2 + 2^n - (C)_2 = 2^n + (B - C)_2$
  - If $B \geq C$
    - $A \geq 2^n$ and the **carry is discarded**.
    - $(A)_2 = (B)_2 + [C]|_{\text{carry discarded}}$
  - If $B < C$
    - $A = 2^n - (C - B)_2 = [C - B]_2$ or $A = -(C - B)_2$ (**no carry in this case**).
  - No overflow for Case 2.

- Example: $(14)_{10} - (9)_{10} = ?$
  - Perform $(14)_{10} + (-(9)_{10})$
  - $(14)_{10} = +(1110)_2 = (0, 1110)_{2cns}$
  - $-(9)_{10} = -(1001)_2 = (1, 0111)_{2cns}$
  - $(14)_{10} - (9)_{10} = (0, 1110)_{2cns} + (1, 0111)_{2cns} = (0, 0101)_{2cns}$ + carry

    $= +(0101)_2 = +(5)_{10}$

Overflow Condition: Case 2 (cont’d)

- Example: $(9)_{10}$ - $(14)_{10}$ = ?
  - Perform $(9)_{10} + (-(14)_{10})$
  - $(9)_{10} = +(1001)_2 = (0, 1001)_{2cns}$
  - $-(14)_{10} = -(1110)_2 = (1, 0010)_{2cns}$
  - $(9)_{10} - (14)_{10} = (0, 1001)_{2cns} + (1, 0010)_{2cns} = (1, 1011)_{2cns}$
    $= -(0101)_2 = -(5)_{10}$

- Example: $(0, 0100)_{2cns} - (1, 0110)_{2cns}$ = ?
  - Perform $(0, 0100)_{2cns} + (-(1, 0110)_{2cns})$
  - $-(1, 0110)_{2cns}$ = two’s complement of $(1,0110)_{2cns}$
    $= (0, 1010)_{2cns}$
  - $(0, 0100)_{2cns} - (1, 0110)_{2cns} = (0, 0100)_{2cns} + (0, 1010)_{2cns}$
    $= (0, 1110)_{2cns} = +(1110)_2 = +(14)_{10}$
  - $+(4)_{10} - (-(10)_{10}) = +(14)_{10}$

Overflow Condition: Case 3

- A = -B - C
  - $A = [B]_2 + [C]_2 = 2^n - (B)_2 + 2^n - (C)_2 = 2^n + 2^n - (B + C)_2 = 2^n + [B + C]_2$
  - Carry bit (2^n) is discarded.
  - An overflow can occur, in which case the sign bit is 0.

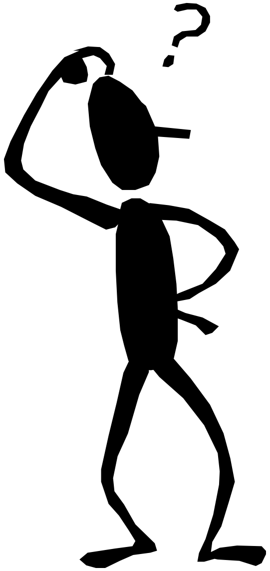
- Example: $-(7)_{10} - (8)_{10} = ?$
  - Perform $(-(7)_{10}) + (-(8)_{10})$
  - $-(7)_{10} = -(0111)_2 = (1, 1001)_{2cns}$, $-(8)_{10} = -(1000)_2 = (1, 1000)_{2cns}$
  - $-(7)_{10} -(8)_{10} = (1, 1001)_{2cns} + (1, 1000)_{2cns} = (11, 0001)_{2cns} = (1, 0001)_{2cns}$ + carry
    $= -(1111)_2 = -(15)_{10}$

- Example: $-(12)_{10} - (5)_{10} = ?$
  - Perform $(-(12)_{10}) + (-(5)_{10})$
  - $-(12)_{10} = -(1100)_2 = (1, 0100)_{2cns}$, $-(5)_{10} = -(0101)_2 = (1, 1011)_{2cns}$
  - $-(12)_{10} -(5)_{10} = (1, 0100)_{2cns} + (1, 1011)_{2cns} =(10, 1111)_{2cns} =(0, 1111)_{2cns}$ + carry
  - Overflow, because the sign bit is 0.

$(1\ 0\ 1111)_{2cns} = -(010001)_2 = -(17)_{10}$

Overflow Condition: Sample

- A = $(25)_{10}$ and B = $-(46)_{10}$
- A + B = ?
- A - B = ?
- B - A = ?
- - A - B = ?



Overflow Condition: Sample (cont'd)

- $A = (25)_{10}$ and $B = -(46)_{10}$
  - $A = +(25)_{10} = (0, 0011001)_{2cns}$
  - $-A = (1, 1100111)_{2cns}$
  - $B = -(46)_{10} = -(0, 0101110)_2 = (1, 1010010)_{2cns}$
  - $-B = (0, 0101110)_{2cns}$
- A+B =
  - $(0, 0011001)_{2cns}+(1, 1010010)_{2cns} = (1, 1101011)_{2cns} = -(21)_{10}$
- A-B =
  - $A+(-B)= (0, 0011001)_{2cns} + (0, 0101110)_{2cns} = (0, 1000111)_{2cns} = +(71)_{10}$
- B-A =
  - $B+(-A)=(1, 1010010)_{2cns}+(1, 1100111)_{2cns}=(1, 0111001)_{2cns}+\textit{carry}=-(0, 1000111)_{2cns}= -(71)_{10}$
- -A-B =
  - $(-A)+(-B)= (1, 1100111)_{2cns} + (0, 0101110)_{2cns} = (0, 0010101)_{2cns} + \textit{carry} = +(21)_{10}$
- Note: Carry bit is discarded.

Overflow Condition: Summary

- Presenting numbers using two's complement number system:
  - Addition: Add two numbers.
  - Subtraction: Add two's complement of the subtrahend to the minuend.
    - **Carry bit is discarded**
  - Overflow is detected as the Table.
  - Radix complement arithmetic can be used for any radix.

Case	Carry	Sign Bit	Condition	Overflow ?
B + C	0	0	$B + C \leq 2^{n-1} - 1$	No
	0	1	$B + C > 2^{n-1} - 1$	Yes
B - C	1	0	$B \leq C$	No
	0	1	$B > C$	No
-B - C	1	1	$-(B + C) \geq -2^{n-1}$	No
	1	0	$-(B + C) < -2^{n-1}$	Yes

Signed Numbers

O Signed and Unsigned Numbers

- Three methods
  - Sign magnitude
  - 2's complement
  - 1's complement

Diminished Radix Complement Number System

- Consider a number $(N)_r$
- Diminished radix complement $[N]_{r-1}$
  - $[N]_{r-1} = r^n - (N)_r - 1$ (1.10)
- One's complement (r = 2):
  - $[N]_{2-1} = 2^n - (N)_2 - 1$ (1.11)
- Example:
  - One's complement of $(01100101)_2$
  - $[N]_{2-1} = 2^8 - (01100101)_2 - 1$

    $= (100000000)_2 - (01100101)_2 - (00000001)_2$

    $= (10011011)_2 - (00000001)_2$

    $= (10011010)_2$

Diminished Radix Complement Number systems (2)

- Example:
  - One's complement of $(01100101)_2$
  - $[N]_{2-1} = 2^8 - (01100101)_2 - 1$

    $= (100000000)_2 - (01100101)_2 - (00000001)_2$

    $= (10011011)_2 - (00000001)_2$

    $= (10011010)_2$

- Example:
  - Nine's complement of (40960)
  - $[N]_{10-1} = 10^5 - (40960)_{10} - 1$

    $= (100000)_{10} - (40960)_{10} - (00001)_{10}$

    $= (59040)_{10} - (00001)_{10}$

    $= (59039)_{10}$

Diminished Radix Complement Number systems (2)

- Find $[N]_{r-1}$ given $(N)_r$.
  - Replace each digit a_i of $(N)_r$ by r - 1 - a.
  - r = 2
    - Simplifies to **complementing** each individual bit of $(N)_r$.
- Radix complement and diminished radix complement of a number (N):
  - $[N]_r = [N]_{r-1} + 1$ (1.12)

Diminished Radix Complement Arithmetic (1)

- Operands are represented in diminished radix complement number system
- Carry is ***added*** to result (end-around carry).
- Example:
  - Add $+(1001)_2$ and $-(0100)_2$.
  - One's complement of +(1001) = 01001
  - One's complement of -(0100) = 11011
  - 01001 + 11011 = 100100

Diminished Radix Complement Arithmetic (1)

- Operands are represented in diminished radix complement number system
- Carry is ***added*** to result (end-around carry).
- Example:
  - Add $+(1001)_2$ and $-(0100)_2$.
  - One’s complement of +(1001) = 01001
  - One’s complement of -(0100) = 11011
  - 01001 + 11011 = 100100 (carry)

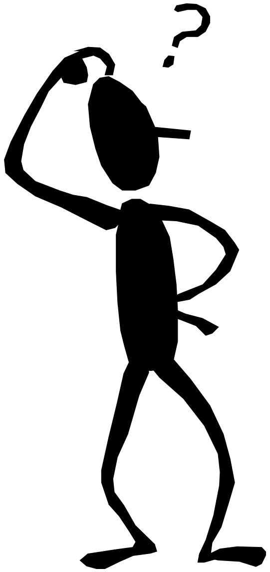
Diminished Radix Complement Arithmetic (1)

دانشگاه علم و صنعت ایران
Iran University of Science & Technology
IUST

- Operands are represented in diminished radix complement number system
- Carry is ***added*** to result (end-around carry).
- Example:
  - Add $+(1001)_2$ and $-(0100)_2$.
  - One's complement of +(1001) = 01001
  - One's complement of -(0100) = 11011
  - 01001 + 11011 = 100100 (carry)
  - ***Add*** the carry to the result: 00100 + 00001
  - correct result is 00101
- Example:
  - Add $+(1001)_2$ and $-(1111)_2$.
  - One's complement of +(1001) = 01001
  - One's complement of -(1111) = 10000
  - 01001 + 10000 = 11001
  - No carry, so this is the correct result
  - -(00110)

Diminished Radix Complement Arithmetic: Sample

- Add $-(1001)_2$ and $-(0011)_2$
- Add $+(75)_{10}$ and $-(21)_{10}$
- Add $+(21)_{10}$ and $-(75)_{10}$



Diminished Radix Complement Arithmetic: Sample (cont'd)

- Add $-(1001)_2$ and $-(0011)_2$
  - One's complement of the operands are:
    - 10110
    - 11100
  - 10110 + 11100 = 110010
    - Carry
  - Correct result is 10010 + 1 = 10011
- Add $+(21)_{10}$ and $-(75)_{10}$
  - Nine's complements of the operands are
    - 021
    - 924
  - 021 + 924 = 945
  - **No carry,** so this is the correct result
- Add $+(75)_{10}$ and $-(21)_{10}$
  - Nine's complements of the operands are:
    - 075
    - 978
  - 075 + 978 = 1053
    - Carry
  - Correct result is 053 + 1 = 054

Thank You